2016 Spring CS300 Homework #4
(Due at May. 16 AM 10:30 on classroom)

**** For all questions, answer as recursive equation or pseudo code is enough****

1. [30pts] Give a solution and analyze it's complexity to the ***0–1 knapsack problem*** that runs in O(n W) time, where n is number of items and W is the maximum weight of items that the thief can put in his knapsack. Value and weight of each item are given as the two sequences v = ⟨v1,v2,...,v*n*⟩ and w = ⟨w1,w2,...,w*n*⟩. (Answer as recursive equation or pseudo code.) Also, prove the correctness of your algorithm.

2. [30pts] In the ***fractional knapsack problem***, the setup is the same, but the thief can take fractions of items, rather than having to make a binary (0-1) choice for each item. Give a solution and analyze it's complexity with Problem 1 arguments (n, W, v, w). Also, prove the correctness of your algorithm.

3. [40pts] A tomato a day Professor Ava loves tomatoes! The professor eats one tomato every day, because she is obsessed with the health benefits of the potent antioxidant lycopene and because she just happens to like them very much, thank you. The price of tomatoes rises and falls during the year, and when the price of tomatoes is low, the professor would naturally like to buy as many tomatoes as she can. Because tomatoes have a shelf-life of only d days, however, she must eat a tomato bought on day i on some day j in the range $i \leftarrow j < i + d$, or else the tomato will spoil and be wasted. Thus, although the professor can buy as many tomatoes as she wants on any given day, because she consumes only one tomato per day, she must be circumspect about purchasing too many, even if the price is low. The professor's obsession has led her to worry about whether she is spending too much money on tomatoes. She has obtained historical pricing data for n days, and she knows how much she actually spent on those days. The historical data consists of an array C[1 . . n], where C[i] is the price of a tomato on day i. She would like to analyze the historical data to determine what is the minimum amount she could possibly have spent in order to satisfy her tomato-a-day habit, and then she will compare that value to what she actually spent. Give an efficient algorithm to determine the optimal offline (20/20 hindsight) purchasing strategy on the historical data. Given d, n, and C[1 . . n], your algorithm should output B[1 . . n], where B[i] is the number of tomatoes to buy on day i.